

Evolutionary Optimization of Functionally Defined Shapes: Case study of Natural Optical Objects

Vladimir V. Savchenko and Alexander A. Pasko

The University of Aizu, Aizu-Wakamatsu City, Fukushima, 965-8580 Japan
{savchen, pasko}@u-aizu.ac.jp

Abstract

This paper focuses on an approach to modeling shapes through the use of evolutionary optimization or genetic algorithms for functionally represented geometric objects. This representation allows us to take into account non-manifold geometry and material distribution inside natural lenses. We use ray tracing as an instrument of the current shape evaluation. We describe a case study of a “four-eyed” fish and discuss problems of using the optimization procedure.

1. Introduction

Computer-aided design (CAD) of optical devices is nowadays an attractive research area with some encouraging examples of applications. Recently, a number of papers were published where authors consider a long-standing problem of CAD of geometric shapes with constraints: to find optimal designs for optical systems [1-3].

Designing a lens by hand or by trial and error is a time-consuming and tedious task. A dream of the end user is to define desired photometric distribution and geometric constraints to design automatically an optical shape satisfying them. Requirements or constraints become more and more difficult to be met using traditional optimization techniques because analytical solutions do not exist for complicated shapes.

In techniques based on the error functional minimization, as noticed by the authors of paper [1], the question may occur to solve highly non-linear problems. Minimization by standard techniques requires high computational effort. Minimization of a simplified functional, for example a quadratic one, is reduced to solving a simple system of linear equations. However, it leads to iterative minimization that depends on a sufficiently good initial guess.

The issue of optimal design for optical systems for functionally represented geometric objects has not yet been addressed. It seems to us that an attractive and a possible way to attack the problem is to use optimization techniques based on genetic algorithms (GAs) as it was shown in book [4].

2. Approach to modeling optical shapes

In this section, we use a known example to introduce and illustrate our shape modeling scheme. A new case study is given in the next section.

2.1 Trilobite eye

Goel and Thompson [4] present an elegant example of using evolutionary optimization technique to create the curved refractive interface in the eye of a certain type of trilobite, the phacopid. Phacopids had eyes containing large biconvex lenses made of durable minerals. The lens consists of two subunits separated by a concave surface. The lens is depicted in Fig. 1. It has a spherical upper surface of radius R_1 and a spherical lower surface (“bowl”) of radius R_2 . Between these two surfaces, there is an intermediate refracting surface that turns the lens shape into a non-manifold object. The region above the lens has a refractive index of $n_1 = 1.33$, and the region below the lens has a refractive index of bodily fluid of $n_4 = 1.34$. The lens subunits have different refractive indexes: $n_2 = 1.66$ for the upper lens and $n_3 = 1.63$ for the lower lens. The authors of [4] assumed that the curved refractive interface in the phacopid eye could be approximated by a sequence of line segments spanning the points (x_i, z_i) , as illustrated in Fig. 1. This unusual shape of the lens allows any light ray passing from point f_1 through the lens to hit focus f_2 . That is why the phacopid could see objects close to it and objects distant from it simultaneously without refocusing. We would like to reconstruct the refracting surface using the

combination of the ray tracing technique, functionally based shape modeling and GAs.

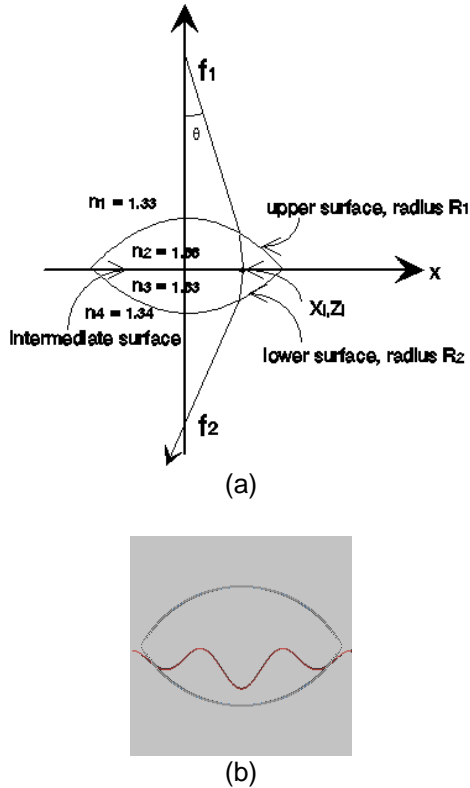


Figure 1. Basic parameters of the trilobite eye (a). Result of evolutionary optimization of the refractive interface (b).

2.2 Shape model

Set-theoretic and many other operations like blending, sweeping, hypertexturing have found quite general solutions for geometric solids represented by continuous real functions as $f(x,y,z) \geq 0$. This representation is called a *function representation* (F-rep) [5]. In paper [6] we presented the state of our volume modeling project mainly based on F-rep and illustrated the approach by several advanced operations on volumetric objects. In spite of well-known problems in using F-rep, in particular the time-consuming evaluation of a defining function, F-rep is robust in floating-point environments and recently we can see increasing interest in the implementation of F-rep in different application areas. There are several reasons for using this representation in optical shape design:

- Shapes can be designed as combinations of analytically defined geometric objects and

volumetric ones that can be obtained from measurements of natural objects.

- Spline-controlled global and local deformations are supported [7].
- The point membership relation based on the three-valued predicate lets us quite easily extend applications to non-manifold objects with intermediate refracting surfaces.
- Parametrized constructive solids (e.g., a lens with holes of variable shapes and positions) can be modeled and optimized.
- Application of the level set methods [8] based on some physically reasonable laws to problems in geometric evolution can be investigated.

With parametrically represented shapes and polygonal meshes, it is difficult to meet some of the above mentioned specifications, for example, to take into account interior distribution of the lens material, to optimize non-manifold or parameterized constructive solids. Also, the shape transformations for such objects suffer from the self-intersection problem. In addition, there is a problem of degenerated polygons as a result of applying deformations.

Pseudocode

```

1. Lens(xt,zt){
2.  shift = Fz(xt);
   //refractive interface
3.  vf1 = sphere(xt,0,zt,R1);
   //upper sphere
4.  vf2 = sphere(xt,0,zt,R2);
   //lower sphere
5.  f_fi = vf1 & vf2;
   // intersection of spheres
6.  f_fo = f_fi & (zt-shift);
   //upper lens unit
7.  f_so = f_fi & (shift-zt);
   //lower lens unit
8.  f_glob = f_fo | f_so;
   //union of two lens units
9.  if(f_fo >= 0.) id_obj = 0;
10. if(f_so >= 0.) id_obj = 1;
11. if(f_glob < 0.) id_obj = 2;
   //Assign a tag for point
   //membership identification
12. return f_glob;}

```

Figure 2. Pseudocode of the functional model of the phacopid eye (see Fig. 1). The intersection and union operations are implemented using R-functions [5]. The cylindrical symmetry of the lens is supposed.

2.3 Shape optimization

To investigate the opportunity of applying an evolutionary optimization technique for CAD of functionally defined optical shapes, we describe in this paper the optimal configuration search for some natural shapes with qualitatively known solutions. In the first example, we test the results of designing curved refractive interface in the phacopid eye described in 2.1. Naturally, approximation of the curved refractive interface by linear segments restricts the applicability of the algorithm [4] to functionally defined shapes. This approximation can cause C^1 discontinuities in the defining function. Using polynomial approximation we have to be cautious about high-order interpolation and errors depending on closeness of the point x to the point x_i . To provide a more smooth surface and a C^1 continuous defining function, we use least square fitting to approximate a sequence of the points (x_i, z_i) , $i = 1, 2, \dots, 12$ by the Fourier series. The unknown refractive interface is described as

$$Fz(x) = b_0 + b_1 \cos(x) + b_2 \cos(2x) + b_3 \cos(3x) + a_1 \sin(x) + a_2 \sin(2x) + a_3 \sin(3x)$$

where

$$b_0 = (z_1 + z_2 + \dots + z_{12})/12$$

$$b_3 = (z_4 - z_2 + z_8 - z_6 + z_{12} - z_{10})/6$$

$$a_3 = (z_1 - z_3 + z_5 - z_7 + z_9 - z_{11})/6$$

$$a_1 = (z_1 - z_9)/2 + a_3$$

$$a_2 = ((z_1 + z_2) - (z_4 + z_5) + (z_7 + z_8) - (z_{10} + z_{11}))/8$$

$$b_1 = (z_0 - z_6)/2 - b_3$$

$$b_2 = (z_0 - z_3 + z_6 - z_9)/4$$

and z_i , $i = 1, 2, \dots, 12$ are coordinates of the unknown refractive interface to be optimized in the points x_i .

The problem of the curved refractive interface design concerns the main feature of the eye to focus a series of light rays emanating at many angles θ_i from a point f_1 to another point f_2 (see Fig.1). The z -intercept of a light ray can be calculated by tracing the ray emanating from f_1 . We can calculate a "fitness" function that expresses the well-known fact that the more effectively the lens can focus lights from f_1 to f_2 the less residual of z -intercepts from the fixed position of focus f_2 we can attain. The pseudocode clarifying the functional description of the lens is shown in Fig. 2.

The calculated tag is used for correct selection of refractive indices in ray tracing. The implemented algorithm is able to start from a flat refractive interface configuration and to generate an optimal set of point coordinates z_i in less than ten minutes of CPU usage on Silicon Graphics Indigo² workstation. As the result of our simulation, we have got the shape (Fig. 1b) actually

qualitatively matching the refractive surfaces designed by the authors of paper [4].

3. Case study of the phaenicophilus palmarum eye

The previous example with modeling the phacopid eye shows some advantages of our approach: simplicity of the functional description of geometric objects, ability to apply quite easily different operations to modify shapes, and the convergence of the optimization method to the acceptable result. Driven by a success with modeling the phacopid eye, we tried to continue consideration of applicability of the combination of ray tracing, optimization techniques based on genetic algorithms and functionally defined geometric objects in design of natural optical shapes.

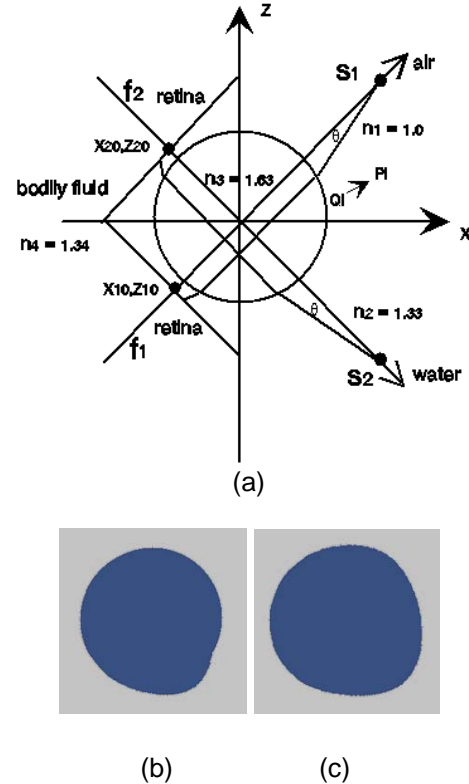


Figure 3. The eye of cuatro ojos, vertical cross section. The eye has two pupils, one looking into the water, the other one into the air (a). Evolutionary optimization of the surface, using four mapping points (b). Evolutionary optimization of the surface, using two mapping points (c).

The main goal of this paper is to show a simple case study that proves that the functional description is a particularly interesting direction for CAD applications, but much work remains to be done before the mentioned above combination becomes a working instrument for the optical design.

The nature gives us the number of very sophisticated examples of shapes. One of them solves an air-water optical problem. Amphibious creatures face a problem if they wish to see both in air and in water. For example, turtles have particularly soft lenses acted on by unusually strong muscles, so they can accommodate for both air and water vision. It seems to us that the easiest way to simulate such soft objects is to use F-rep. Sobel in his book [9] describes an exotic natural solution to the air-water vision problem. The phaenicophilus palmarum (cuatro ojos in [9]), or the “four-eyed” fish, swims along the water surface with its head partly out of water, and each of its two eyes has two pupils, one kept above the surface, the other below. Each pupil focuses light upon a different retina (Fig. 3). There is only one lens, approximately of elliptical shape, cleverly oriented so that the pupil in water focuses through the more sharply curved surface of the lens, while the pupil in air focuses through the less sharply curved surface. The fish thus obtains two focused images, one of the sky and one of the sea.

We consider a simple model of a “four-eyed” fish. Here, we also use the fact that one of the key functions of the eye is to focus a series of rays θ_i ($i = 1, 2, \dots, k$, k - number of rays) emanating at many angles from source points S_1, S_2 to focusing points f_1, f_2 (see Fig. 3). The focusing points are defined close to the lens and are supposed to be on the retina. The $x_{1i} z_{1i}, x_{2j} z_{2j}$ -intercepts are coordinates of light rays that have passed through the lens after living starting points and can be calculated by tracing the rays through the lens surface according to the Snell’s law where we use the refractive indices of n_1, n_2, n_3, n_4 for the sea water, air, lens and bodily fluid respectively.

We consider space mappings as a main tool for the shape transformation. A space mapping establishes one-to-one correspondence between points of the space and is defined by the functional dependence between new and old coordinates of control points. We apply space mappings driven by control points linked to an object [7] for soft shape transformations.

A space mapping in \mathbb{R}^n defines the relationship between each point in the original and deformed object. Let an n -dimensional region of an arbitrary configuration be given and contains a set of arbitrary control points $\{Q_i = (q_1^i, q_2^i, \dots, q_n^i) : i=1, 2, \dots, M\}$ for a

nondeformed object and $\{P_i = (p_1^i, p_2^i, \dots, p_n^i) : i=1, 2, \dots, M\}$ for a deformed object. By assumption, the points Q_i and P_i are distinct and given on or near a surface of an object. The goal of the construction of the deformed object is to find a smooth mapping function that approximately describes the spatial transformation. The inverse mapping function can be given in the form:
 $Q_i = U(P_i) + P_i$

where the components of the vector $U(P_i)$ are the volume splines interpolating displacements of initial points Q_i . Note, since the algorithm does not use any regular grid, M control points can be freely chosen by the user as initial approximations for deformation.

We define the “fitness” function as follows

$$F(P_1, P_2, \dots, P_M) = 1 - \sum_{i=1}^k d_{1i} w_w - \sum_{j=1}^l d_{2j} w_a$$

where $d_{1i} = \sqrt{(x_{10} - x_{1i})^2 + (z_{10} - z_{1i})^2}$ and $d_{2j} = \sqrt{(x_{20} - x_{2j})^2 + (z_{20} - z_{2j})^2}$. Such definition of the fitness function expresses the same idea that the more effectively the lens can focus lights from S_1 and S_2 the less scattering of lights can be achieved on the retina. Also, we take into account the number of light rays passing the retina from air and water with the help of weighting coefficients w_a, w_w .

Pseudocode

```

1. Lens(x, z) {
2. (xt, zt) = calc_spline(x, z)
//point transformation
3. f_glob = sphere(xt, 0, zt, R);
//transformed sphere
4. if(f_glob < 0.) id_obj = 2;
5. if(z > 0.) id_obj = 1;
6. if(f_glob >= 0.) id_obj = 0;
//Assign a tag for point
//membership identification
7. return f_glob;

```

Figure 4. Pseudocode of the functional model of the cuatro ojos eye

The pseudocode is provided in Fig. 4 to clarify the functional description of the deformed lens and tag assignment for the point membership identification. A spherical lens shape is taken as an initial shape. We consider a 2D vertical cross section as it is described in [9].

We simulate the deformation by random assigning starting coordinates of control points $\{P_i = (p_1^i, p_2^i, \dots, p_n^i) : i=1, 2, \dots, M\}$ for a deformed object and apply GA to minimize the “fitness” function. Essentially, GA is a form of artificial evolution and is

similar to the trial-and-error approach. The most attractive feature of GAs is that they are easy to interface to a model. The main drawback of GAs is that the optimization problem can be trapped in a local minimum. We apply the simplest genetic algorithm developed by Holland et al. [10]. The algorithm begins with a set of random structures as possible solutions to a given problem. The structures (P_i coordinates) are evaluated using a fitness weighted random selection scheme, and applying genetic operators such as mutation and crossover to them. The resulting deformed shape is then evaluated with the help of the recursive ray tracing algorithm which allows us to take into account a collection of refracted and reflected rays and the new structures with higher fitness join the population to replace those old ones whose fitness measures are lower.

4. Discussion

Our experiments show that with the evolutionary optimization we can design a shape of a “four-eyed” fish matching the natural one according to the qualitative description given in [9]. The pupil looking into the water uses the more sharply curved surface of the lens to help converge light on the retina. The pupil looking into the air uses the less sharply curved surface of the lens (Fig. 3 b, c).

Unfortunately, the experiments showed that we have the problem with the algorithm convergence. The authors of paper [4] noticed that the problem of local trapping usually arises when the optimization process is forced to make a decision between different ways of increasing fitness, for example, if many variables in the system are allowed to be free. In our experiments, the optimization procedure is able to converge to the optimal solution with a very slow rate if we increase the number of mapping points. We noticed that there is a strong tendency for the system to become more fit if we use only two mapping points (Fig. 3c). In our case we have an additional problem as we try to find a solution to satisfy two different focuses in accordance with one fitness criterion as discussed above. In our opinion, the convergence problem may arise in view of “badly” defined fitness criteria and finding some “good” fitness criteria is an open question. Fortunately, as mentioned in paper [11], an algorithm can have significant practical value even if it fails to identify the true global optimal solution because the algorithm can find alternative solutions that more fully satisfy some minimal set of user-defined requirements.

In spite of the above mentioned problems, we believe that a good alliance of geometric modeling and optimization techniques can be found to prove the

applicability of GAs in optical design and consider the modeling example as a challenging problem not only for researchers from CG area but we hope that those who are working in the rapidly growing area of genetic algorithms could give some help to us. From the geometrical point of view we consider the function representation a good candidate for such modeling despite the fact that this technology is still in its infancy.

References

- [1] J. Loos, Ph. Slusallek, H.-P. Seidel, Using wavefront tracing for the visualization and optimization of progressive lenses, *EUROGRAPHICS'98*, Computer Graphics Forum, vol.17, No. 3, 1998, 255-265.
- [2] J. Loos, G. Greiner, H.-P. Seidel, A variational approach to progressive lens design, *Computer-Aided Design*, vol. 30, No. 8, 1998, 595-602.
- [3] M. Halstead, B. Barsky, S. Klein, R. Mandell, Reconstructing curved surfaces from specular reflection patterns using spline surface fitting of normals. *SIGGRAPH 96 Conference Proceedings*, H. Rushmeier ed., Addison Wesley, August 1996, 335-342.
- [4] M. Goel, R.L. Thompson, The evolution of the trilobite eye, *Computer Simulations of Self-Organization in Biological Systems*, part 11, Croom Helm, London & Sydney, 1988, 275-290.
- [5] A. Pasko, V. Adzhiev, A. Sourin, V. Savchenko, Function representation in geometric modeling: concepts, implementation and applications, *The Visual Computer*, vol.11, No.8, 1995, 429-446.
- [6] V.V. Savchenko, A.A. Pasko, A.I. Surin, T.L. Kunii, Volume modeling: representations and advanced operations, *Proceedings of CGI '98*, Hannover, Germany, June 22-26, 1998, 4-13.
- [7] V.V. Savchenko, A.A. Pasko, Transformation of functionally defined shapes by extended space mappings, *The Visual Computer*, vol. 14, Nos. 5/6, 1998, 257-270.
- [8] J.A. Sethian, *Level Set Methods*, Cambridge University Press, 1996.
- [9] M.I. Sobel, *Light*, The University of Chicago Press, Ltd., London, 1987, p. 263.
- [10] J.H. Holland, K.J. Holyoak, R.E. Nisbett, P.R. Thagard, *Induction: Processes of Inference Learning and Discovery*, MIT Press, Cambridge, MA, 1986.
- [11] T. Kuper, Global optimization finds alternative lens design, *Laser Focus World*, vol. 28, No. 5, 1992, 193-195.